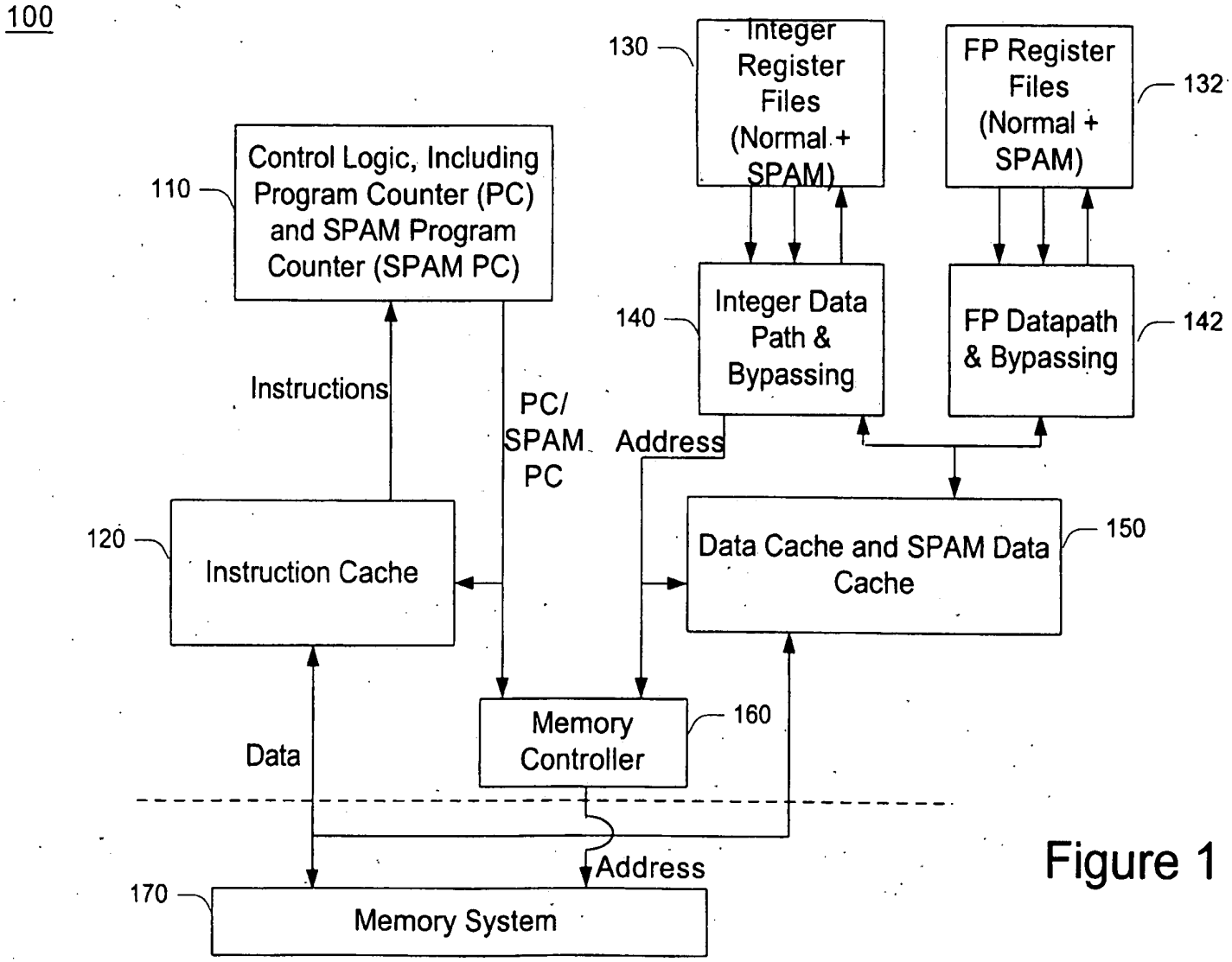
100
130
Integer Register Files (Normal + SPAM)
132
FP Register Files (Normal + SPAM)
110
Control Logic, Including Program Counter (PC) and SPAM Program Counter (SPAM PC)
140
Integer Data Path & Bypassing
142
FP Datapath & Bypassing
Instructions
PC/ SPAM PC
Address
120
Instruction Cache
150
Data Cache and SPAM Data Cache
160
Memory Controller
Data
Address
170
Memory System

Figure 1

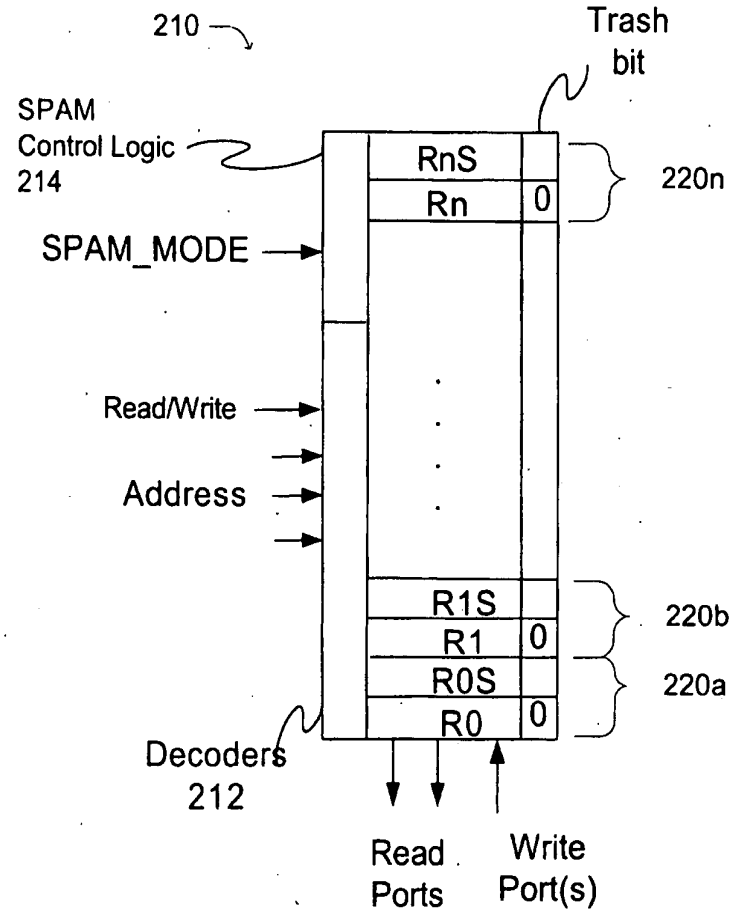

Figure 2A

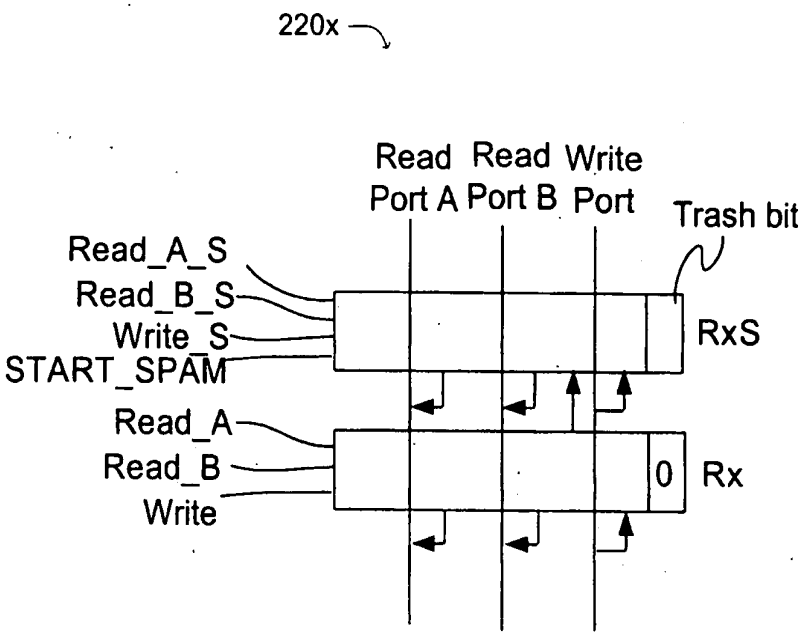

Figure 2B

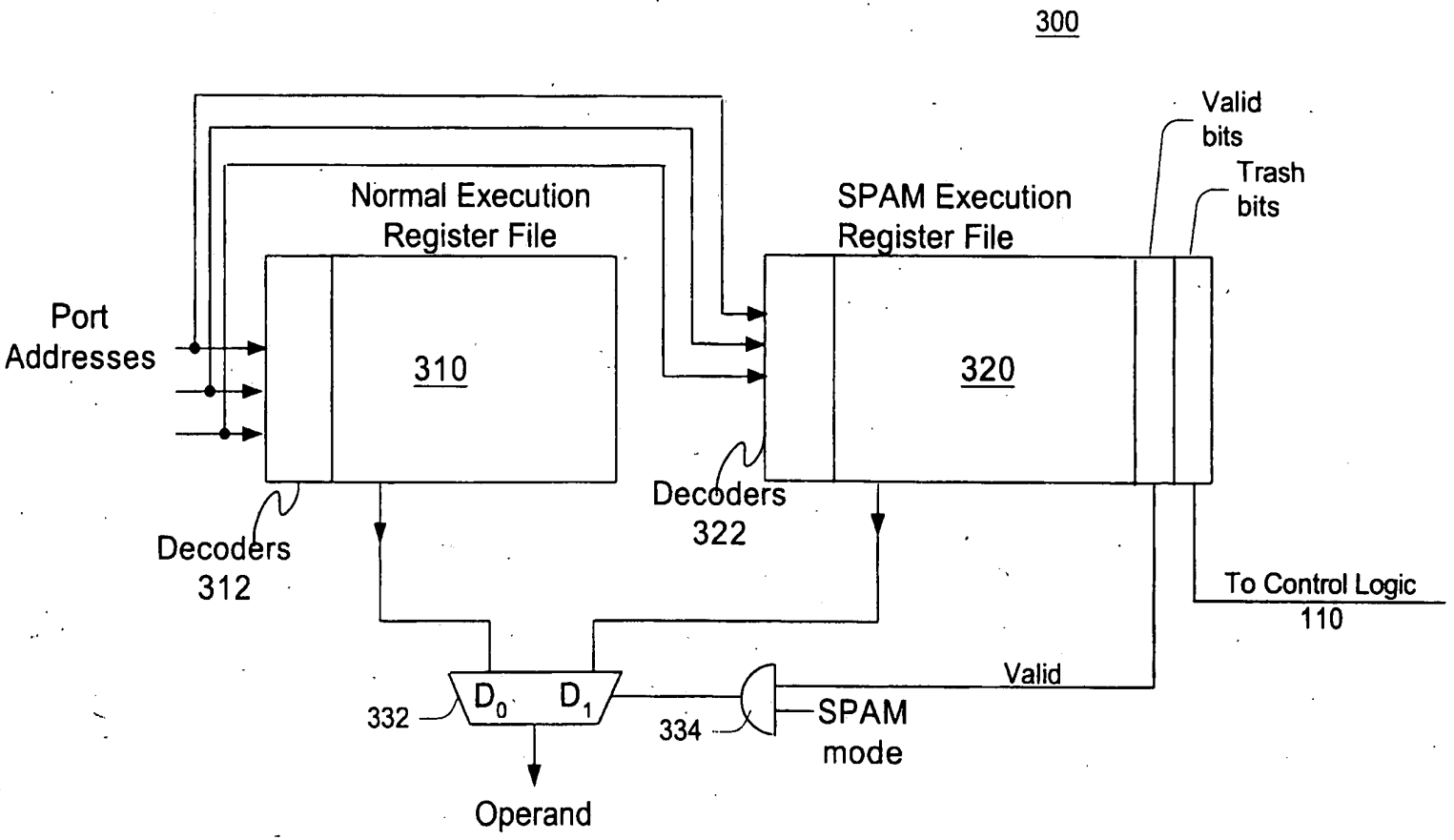
300
Valid
bits
Trash
bits
Normal Execution
Register File
SPAM Execution
Register File
Port
Addresses
310
320
Decoders
322
Decoders
312
To Control Logic
110
Valid
D_0
D_1
332
334
SPAM
mode
Operand

Figure 3A

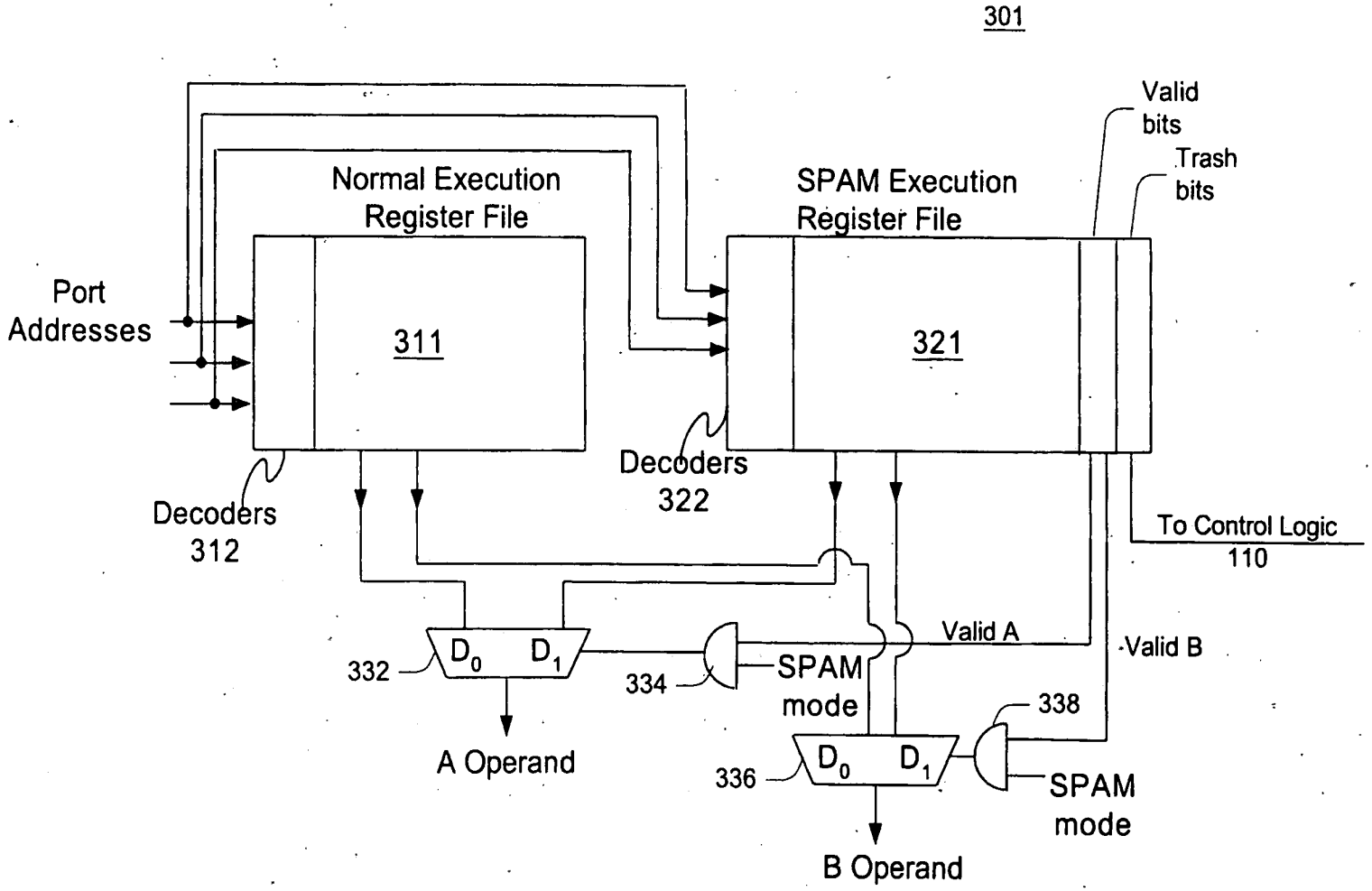
301
Normal Execution
Register File
SPAM Execution
Register File
Valid
bits
Trash
bits
Port
Addresses
311
321
Decoders
312
Decoders
322
To Control Logic
110
Valid A
Valid B
332
D_0 D_1
334
SPAM
mode
338
A Operand
336
D_0 D_1
SPAM
mode
B Operand

Figure 3B

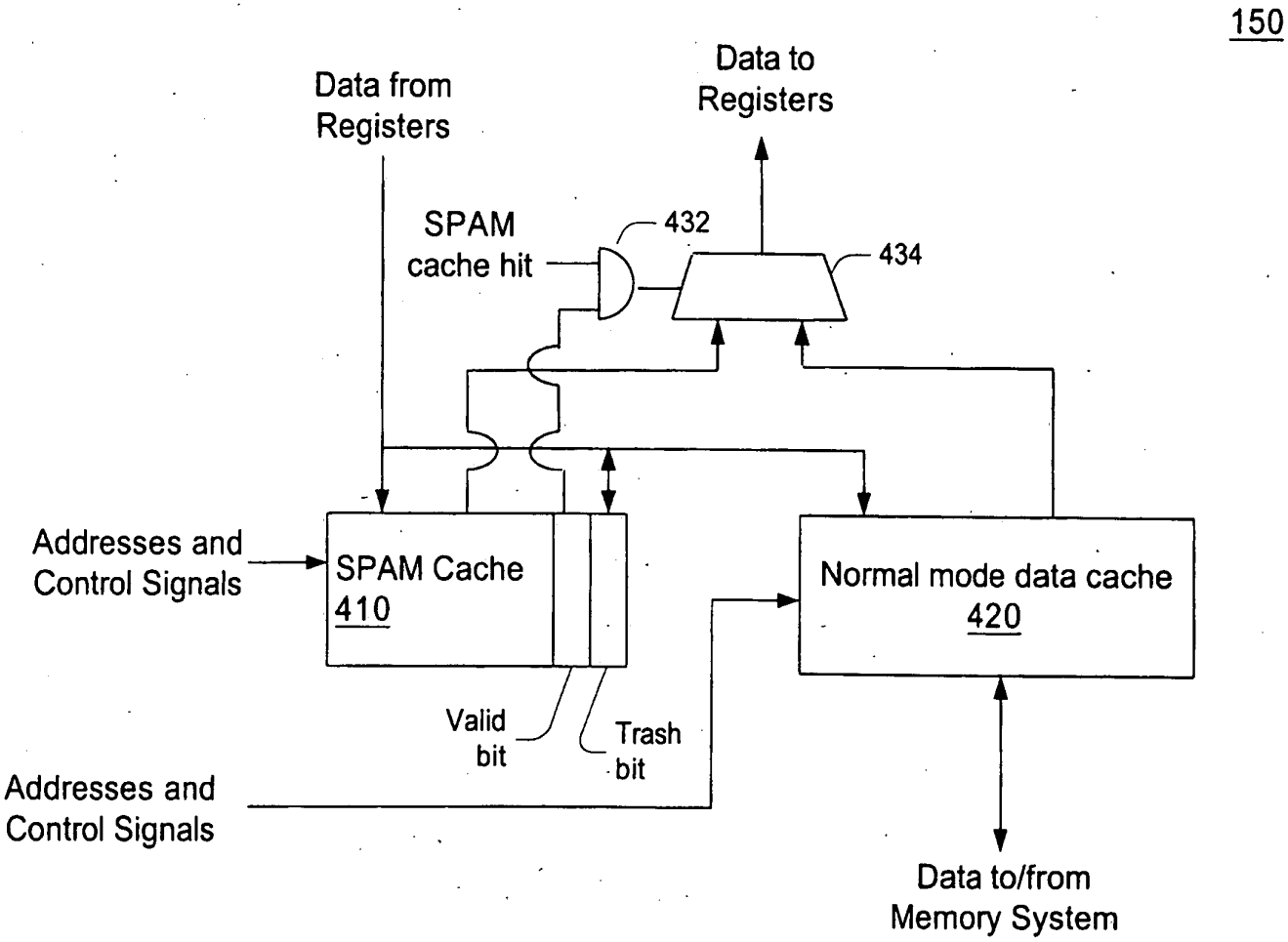
150
Data to
Registers
Data from
Registers
SPAM
cache hit
432
434
Addresses and
Control Signals
SPAM Cache
410
Normal mode data cache
420
Valid
bit
Trash
bit
Addresses and
Control Signals
Data to/from
Memory System

Figure 4

500

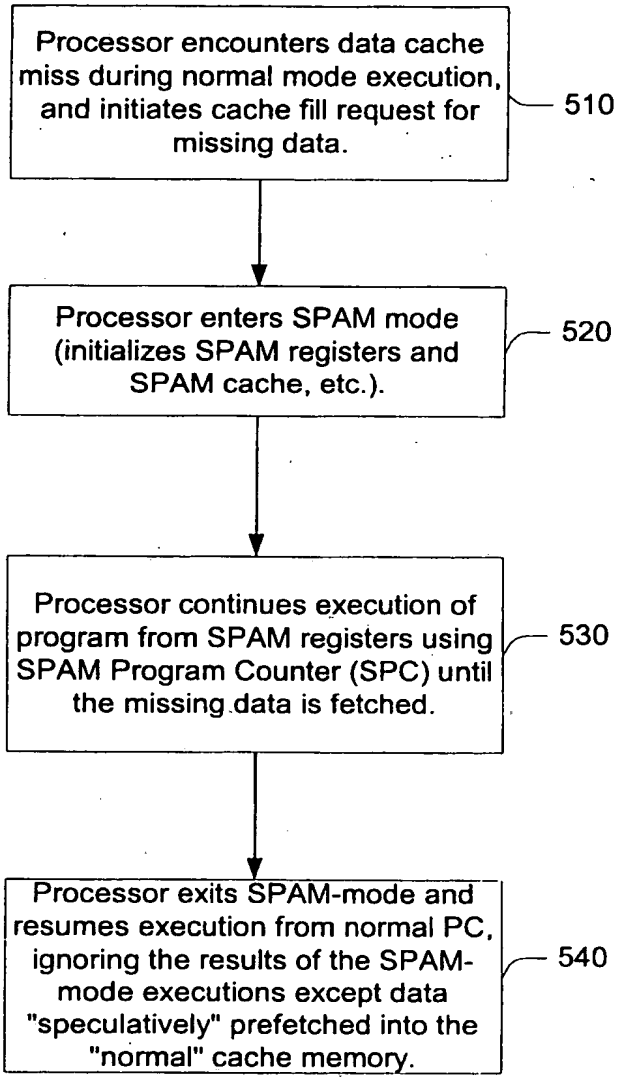
Processor encounters data cache miss during normal mode execution, and initiates cache fill request for missing data.
510
Processor enters SPAM mode (initializes SPAM registers and SPAM cache, etc.).
520
Processor continues execution of program from SPAM registers using SPAM Program Counter (SPC) until the missing data is fetched.
530
Processor exits SPAM-mode and resumes execution from normal PC, ignoring the results of the SPAM-mode executions except data "speculatively" prefetched into the "normal" cache memory.
540

Figure 5

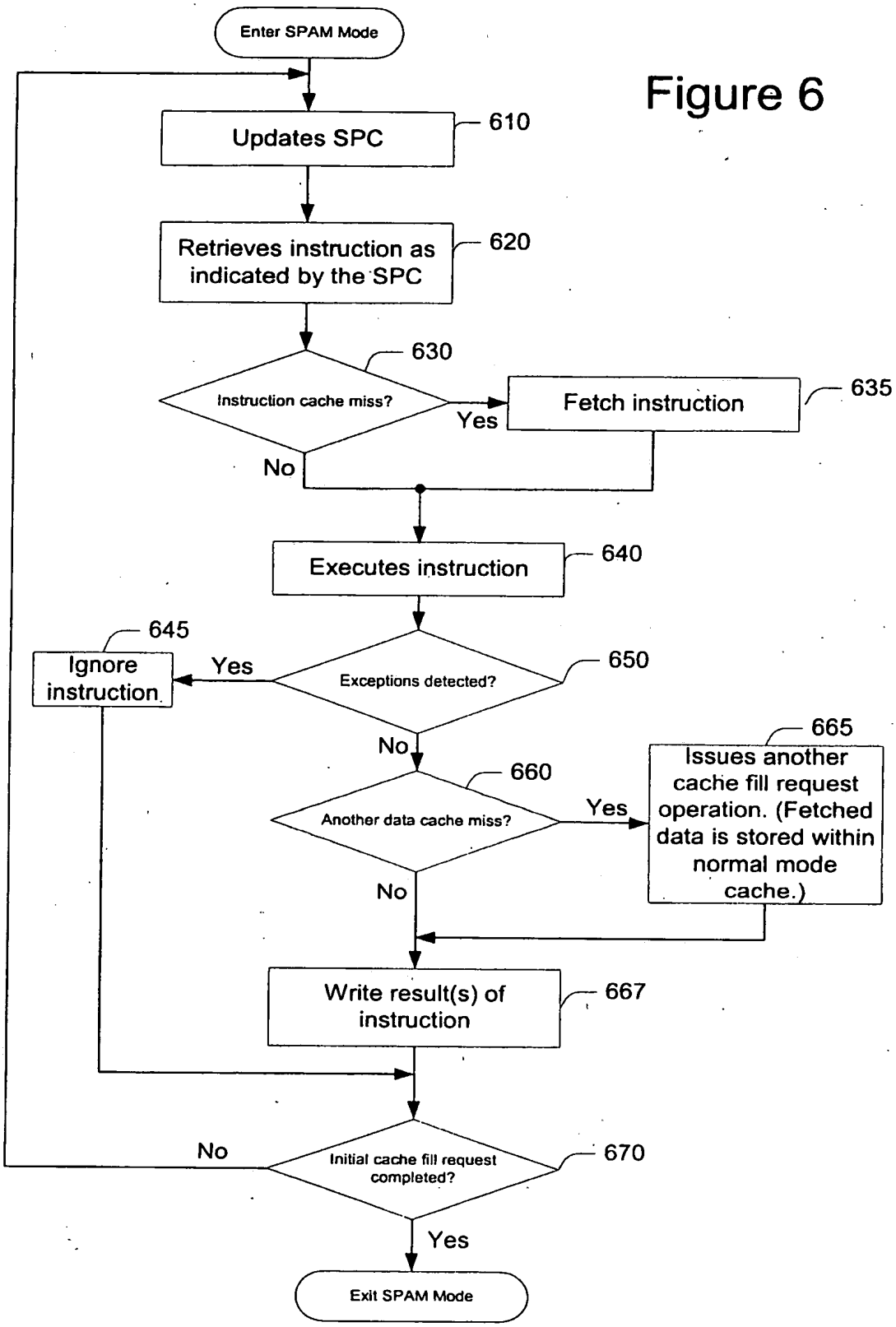
Enter SPAM Mode
Updates SPC
610
Retrieves instruction as indicated by the SPC
620
630
Instruction cache miss?
Yes
Fetch instruction
635
No
Executes instruction
640
645
Ignore instruction
Yes
Exceptions detected?
650
No
665
Issues another cache fill request operation. (Fetched data is stored within normal mode cache.)
660
Another data cache miss?
Yes
No
Write result(s) of instruction
667
No
Initial cache fill request completed?
670
Yes
Exit SPAM Mode

Figure 6

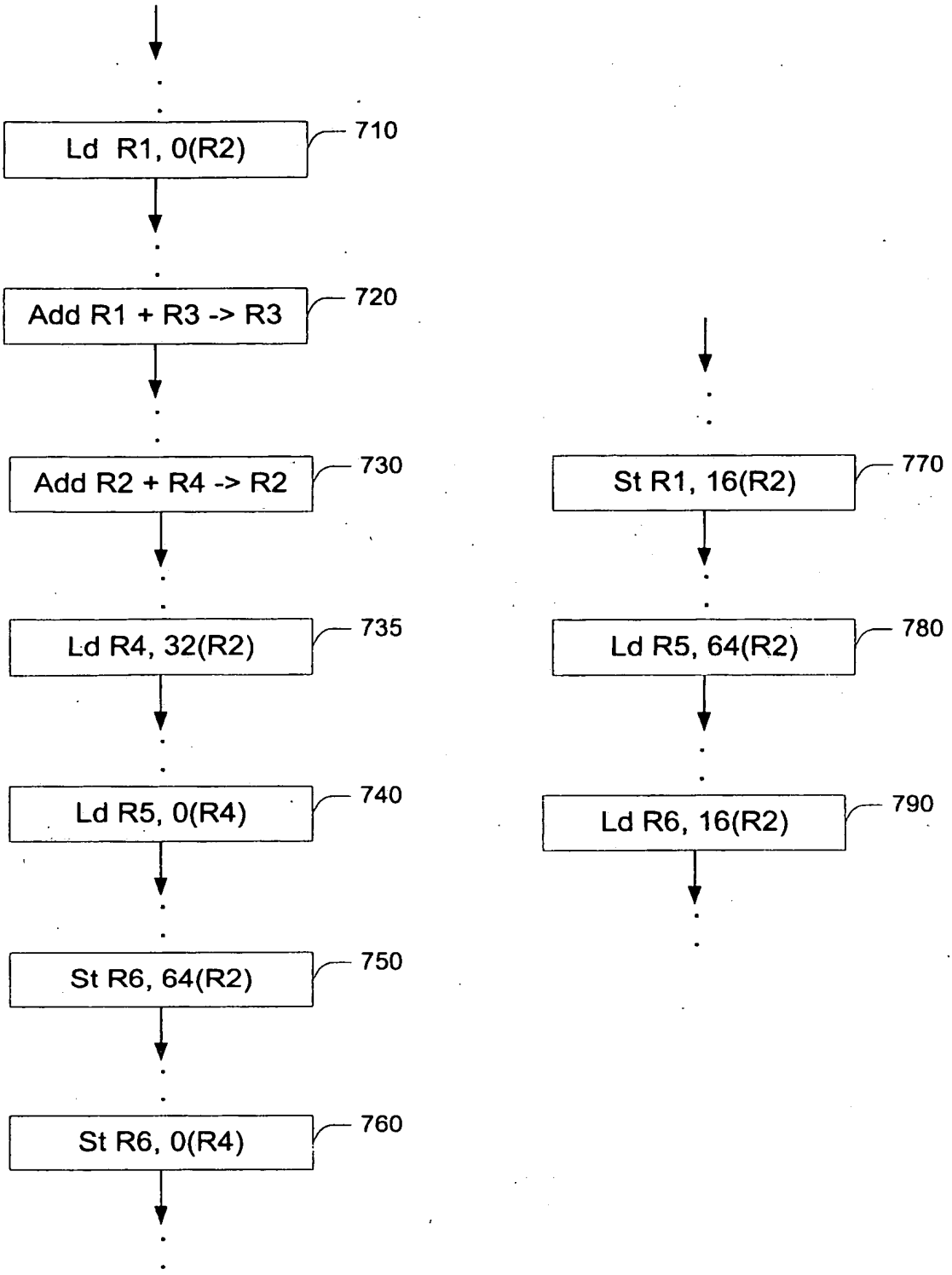
Ld R1, 0(R2)
710
Add R1 + R3 -> R3
720
Add R2 + R4 -> R2
730
Ld R4, 32(R2)
735
Ld R5, 0(R4)
740
St R6, 64(R2)
750
St R6, 0(R4)
760
St R1, 16(R2)
770
Ld R5, 64(R2)
780
Ld R6, 16(R2)
790

Figure 7

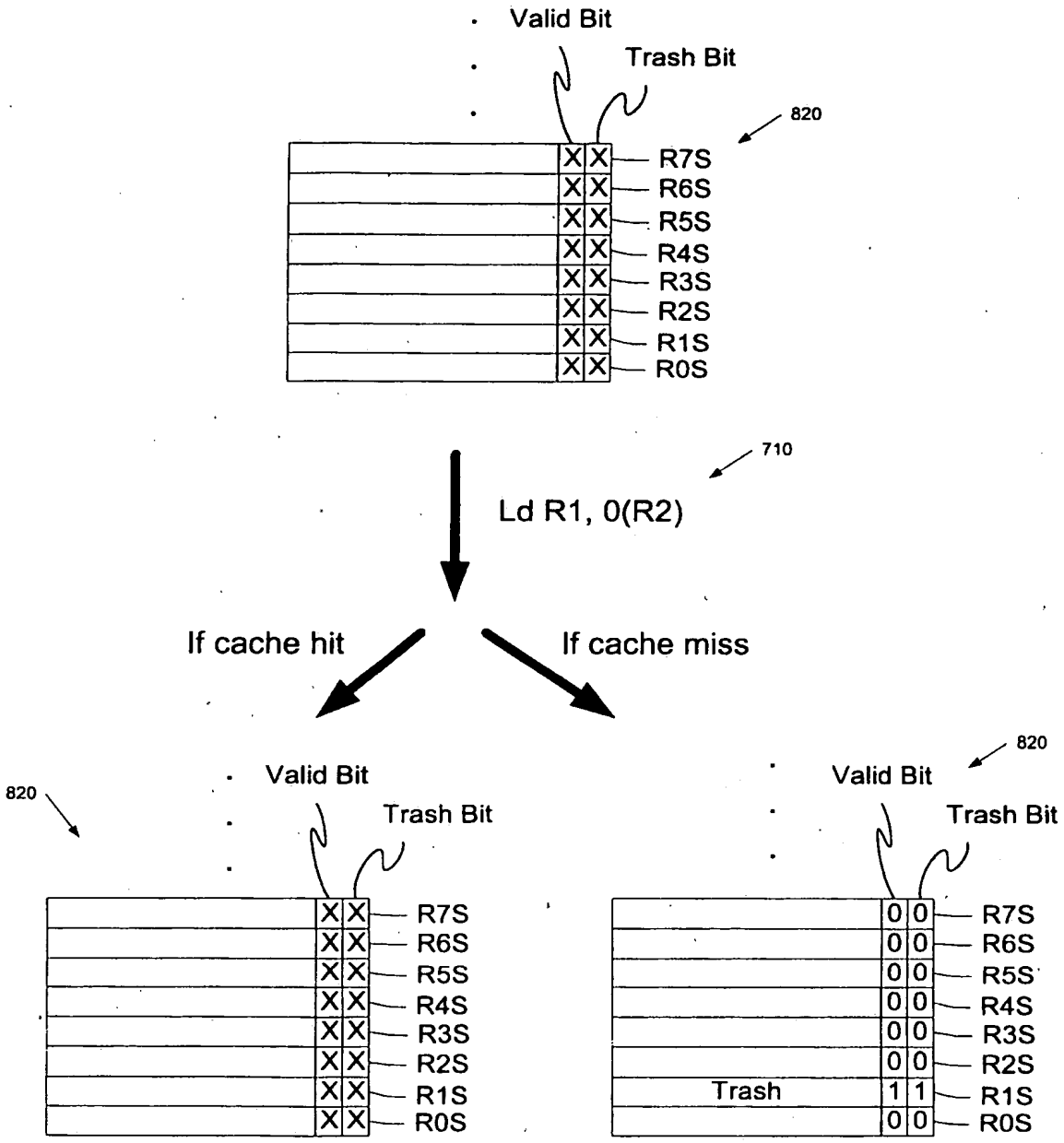
Valid Bit
Trash Bit
820
X X R7S
X X R6S
X X R5S
X X R4S
X X R3S
X X R2S
X X R1S
X X R0S
710
Ld R1, 0(R2)
If cache hit
If cache miss
820
Valid Bit
Trash Bit
X X R7S
X X R6S
X X R5S
X X R4S
X X R3S
X X R2S
X X R1S
X X R0S
820
Valid Bit
Trash Bit
0 0 R7S
0 0 R6S
0 0 R5S
0 0 R4S
0 0 R3S
0 0 R2S
Trash 1 1 R1S
0 0 R0S

Figure 8

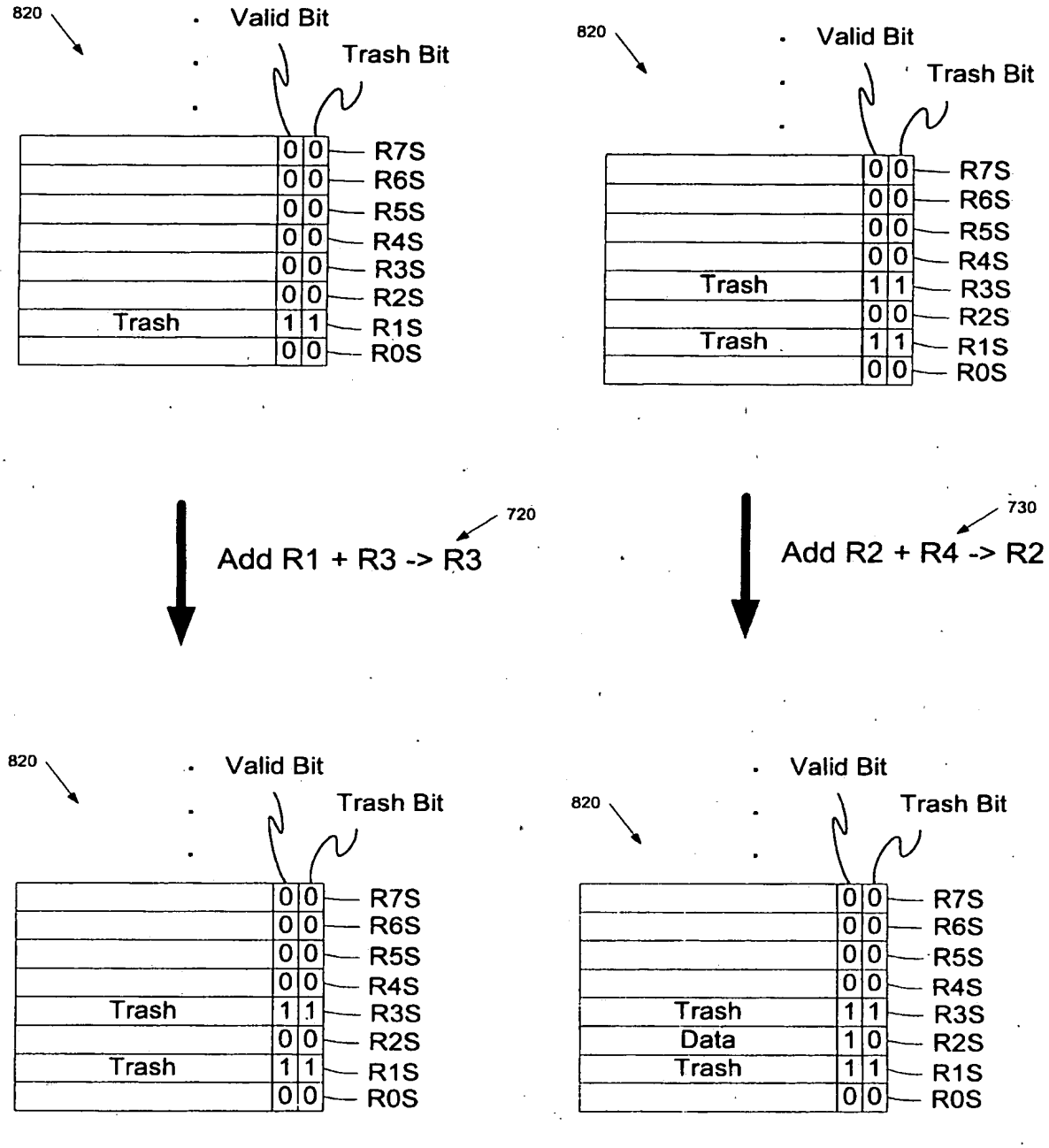

Figure 9A

Figure 9B

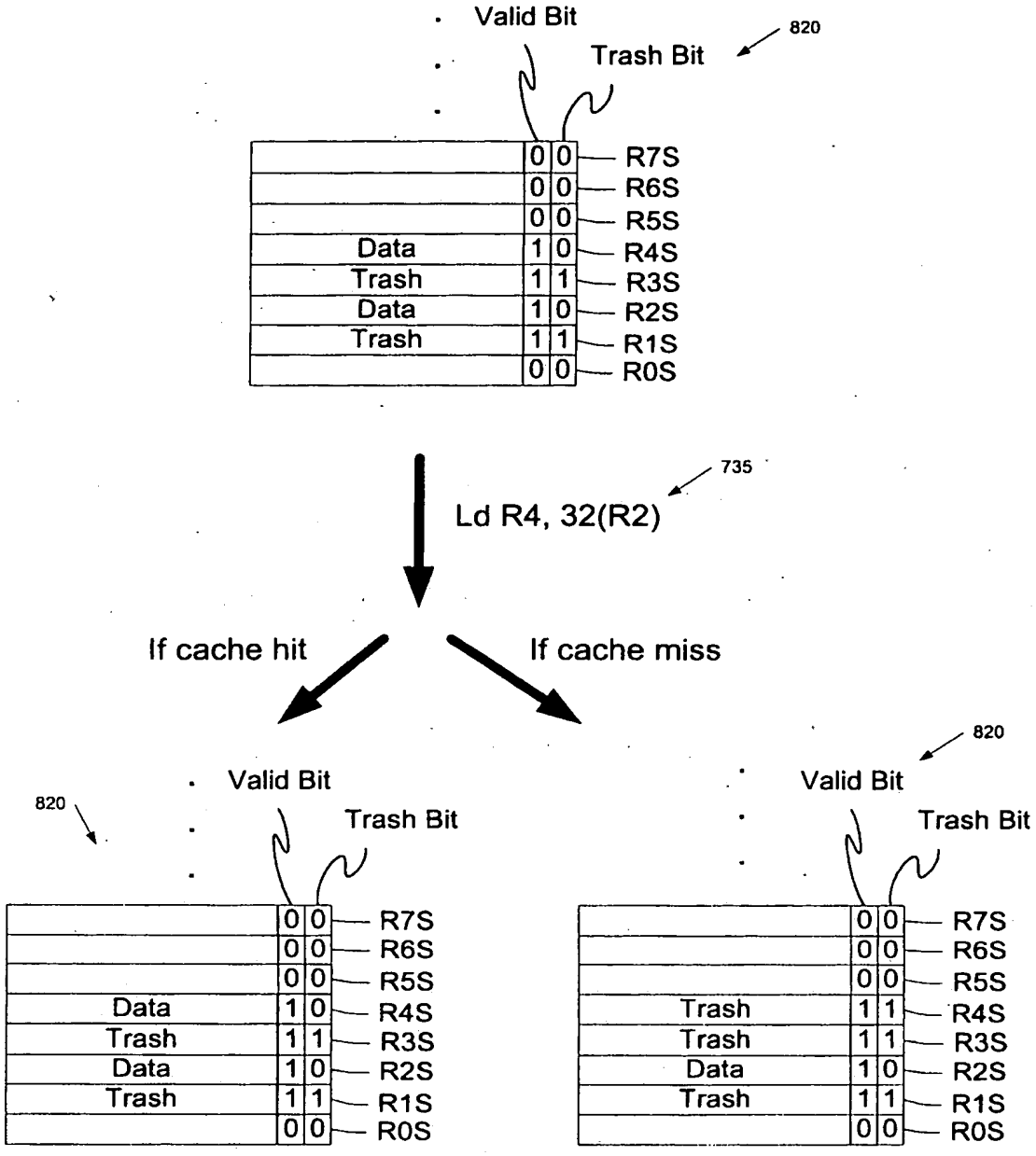
Valid Bit
Trash Bit
820
R7S
R6S
R5S
Data
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S
735
Ld R4, 32(R2)
If cache hit
If cache miss
820
Valid Bit
Trash Bit
R7S
R6S
R5S
Data
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S
820
Valid Bit
Trash Bit
R7S
R6S
R5S
Trash
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S

Figure 10

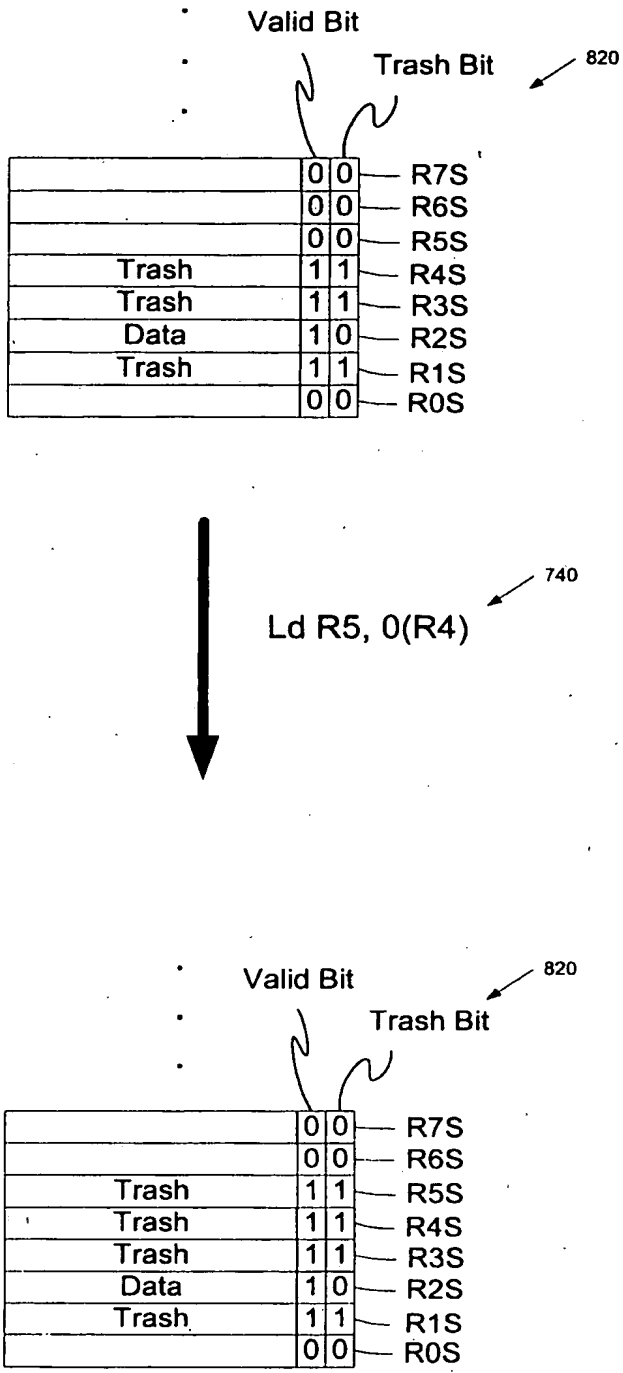
Valid Bit
Trash Bit
820
R7S
R6S
R5S
R4S
R3S
R2S
R1S
R0S
Trash
Trash
Data
Trash
740
Ld R5, 0(R4)
Valid Bit
Trash Bit
820
R7S
R6S
R5S
R4S
R3S
R2S
R1S
R0S
Trash
Trash
Trash
Data
Trash

Figure 11

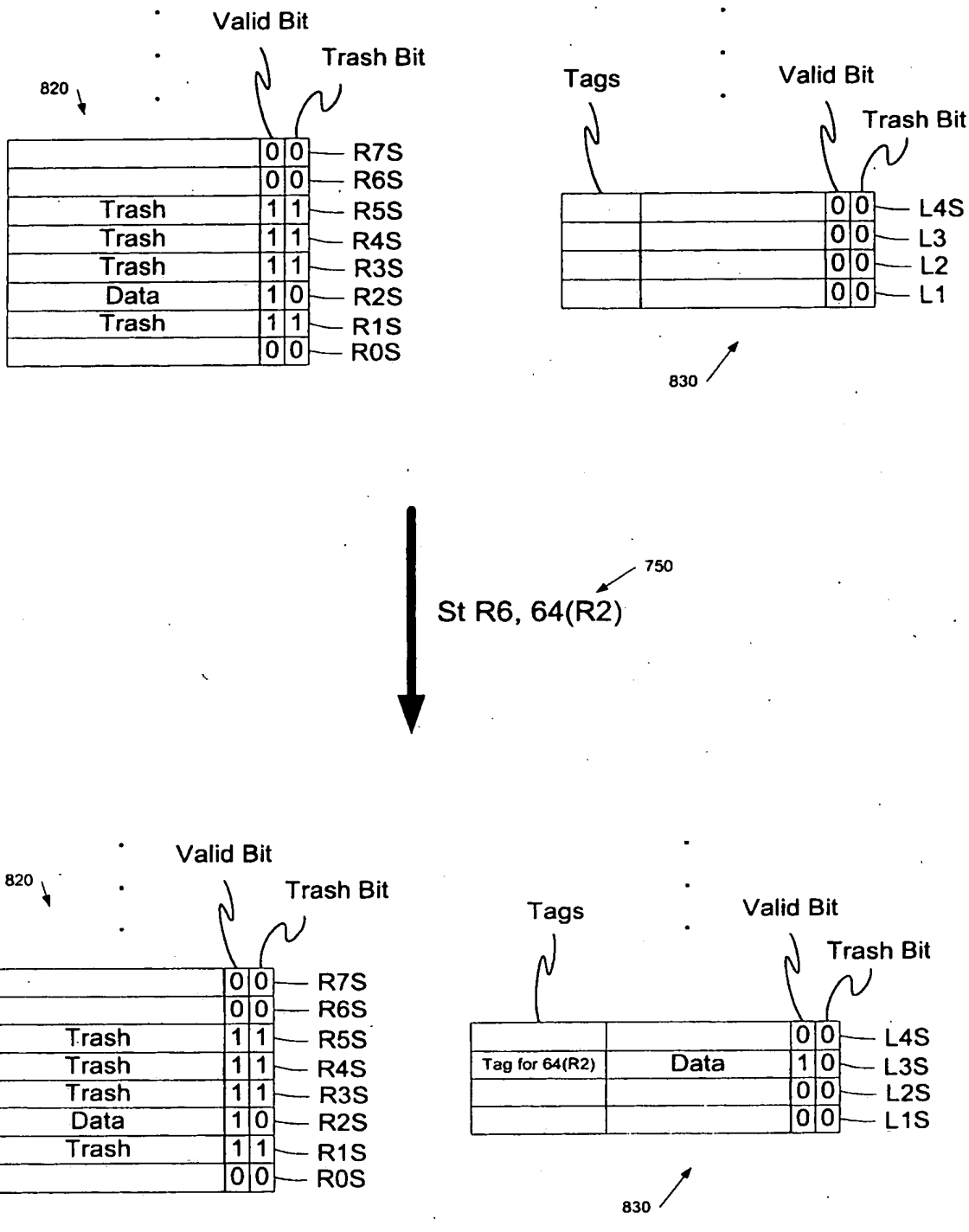
Valid Bit
Trash Bit
820
R7S
R6S
Trash
R5S
Trash
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S
Tags
Valid Bit
Trash Bit
L4S
L3
L2
L1
830
750
St R6, 64(R2)
Valid Bit
Trash Bit
820
R7S
R6S
Trash
R5S
Trash
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S
Tags
Valid Bit
Trash Bit
Tag for 64(R2)
Data
L4S
L3S
L2S
L1S
830

Figure 12

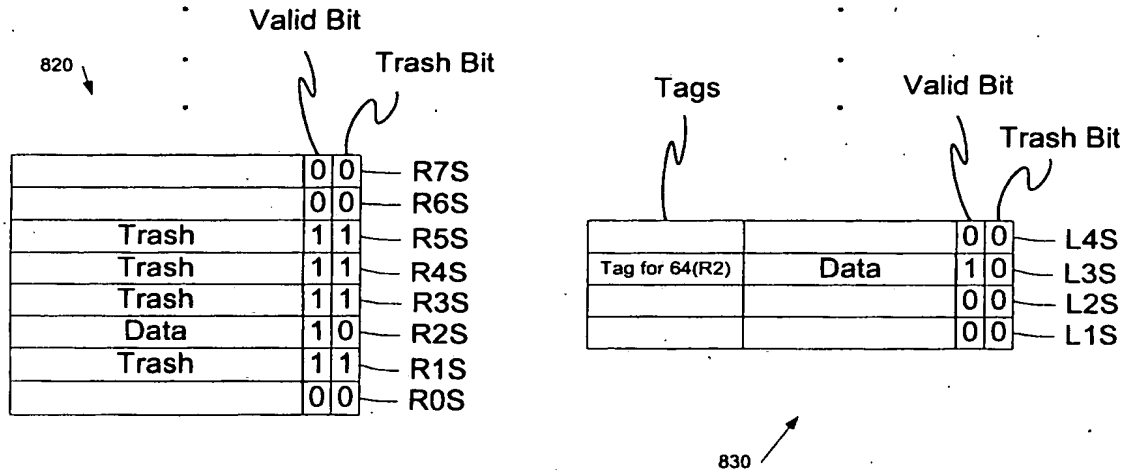
820
Valid Bit
Trash Bit
0 0 R7S
0 0 R6S
Trash 1 1 R5S
Trash 1 1 R4S
Trash 1 1 R3S
Data 1 0 R2S
Trash 1 1 R1S
0 0 R0S
Tags
Valid Bit
Trash Bit
0 0 L4S
Tag for 64(R2) Data 1 0 L3S
0 0 L2S
0 0 L1S
830
760
St R6, 0(R4)
No - op

Figure 13

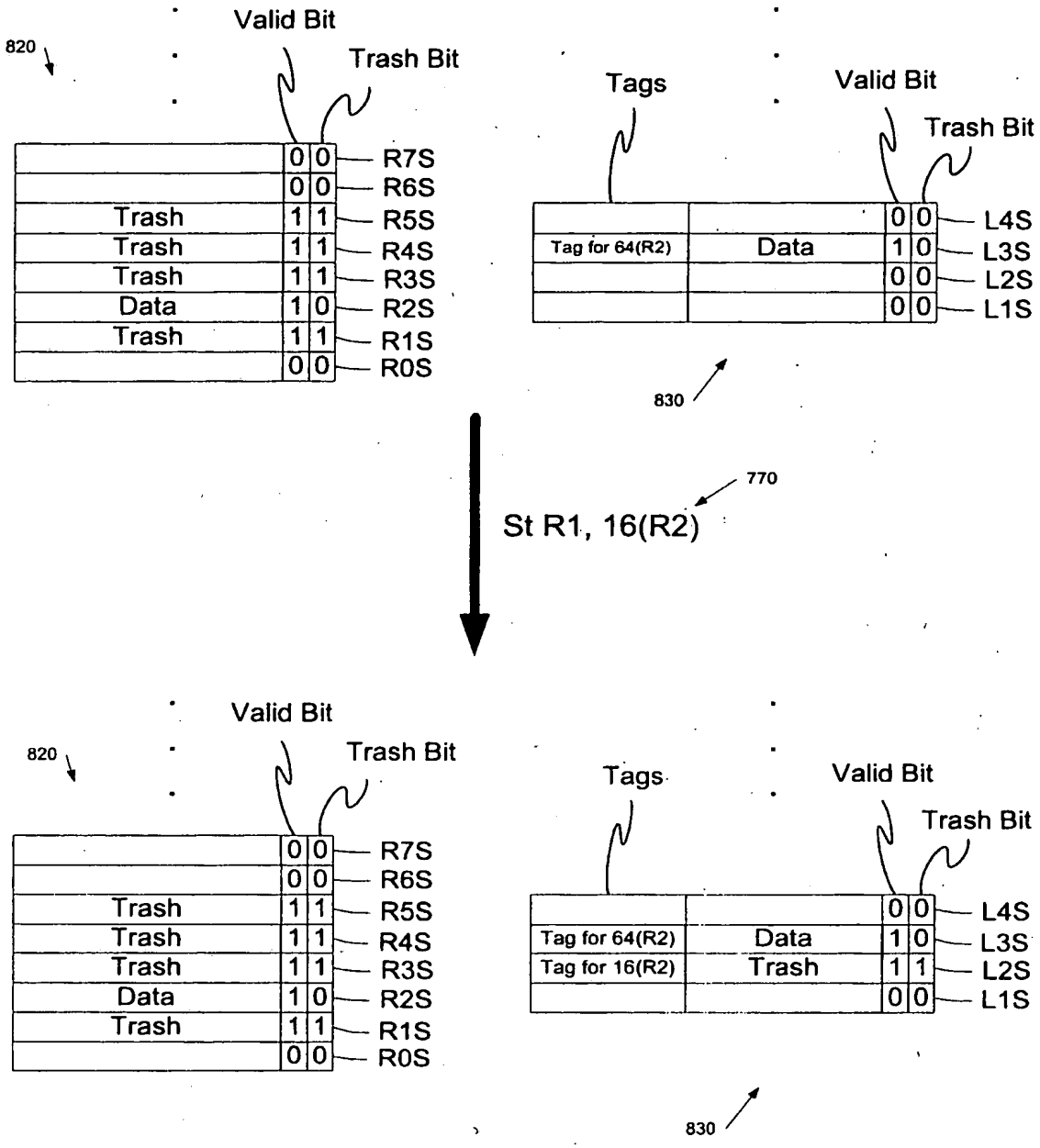
820
Valid Bit
Trash Bit
R7S
R6S
Trash
R5S
Trash
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S
Tags
Valid Bit
Trash Bit
Tag for 64(R2)
Data
L4S
L3S
L2S
L1S
830
770
St R1, 16(R2)
820
Valid Bit
Trash Bit
R7S
R6S
Trash
R5S
Trash
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S
Tags
Valid Bit
Trash Bit
Tag for 64(R2)
Data
Tag for 16(R2)
Trash
L4S
L3S
L2S
L1S
830

Figure 14

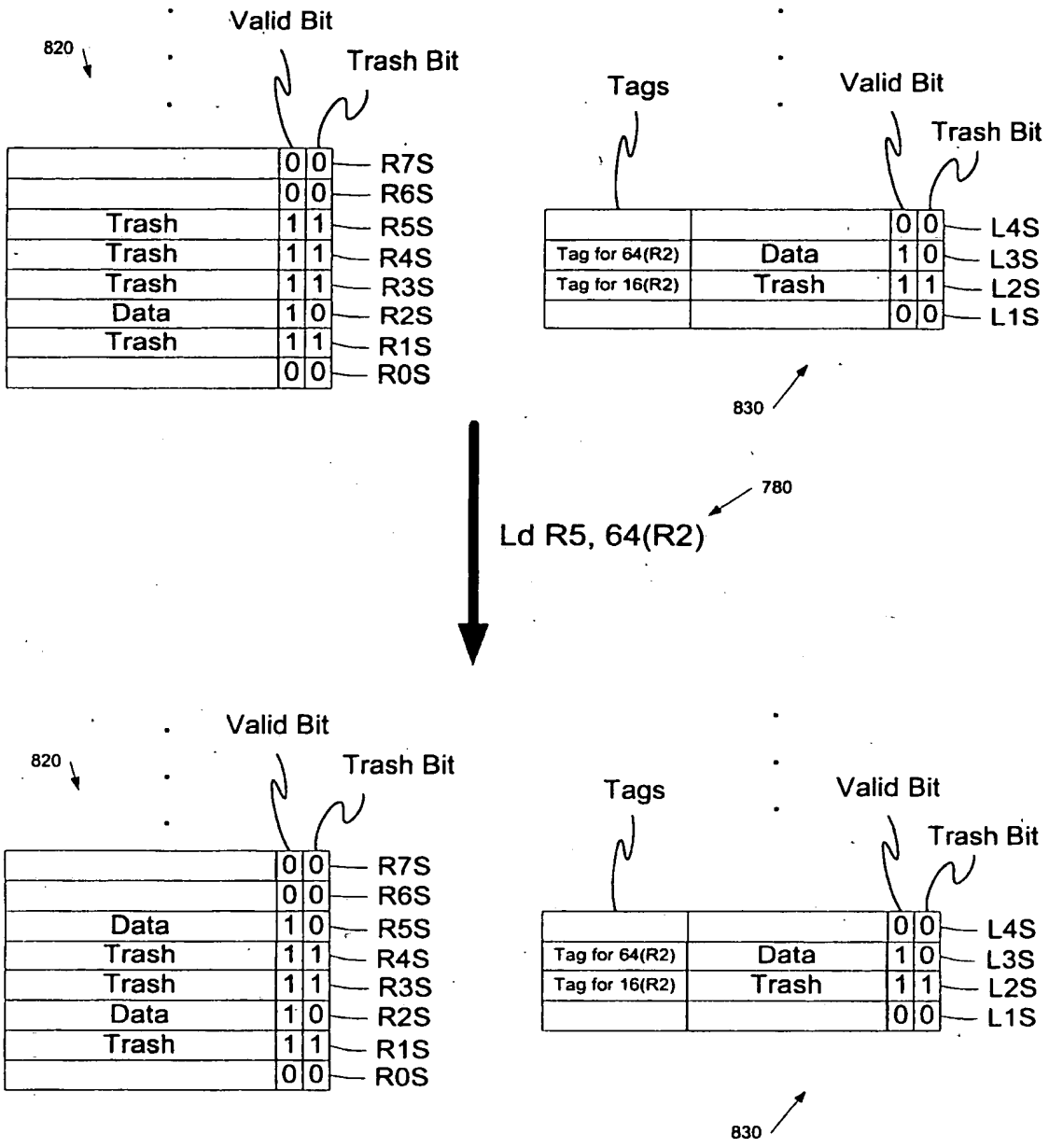
820
Valid Bit
Trash Bit
R7S
R6S
Trash
R5S
Trash
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S
Tags
Valid Bit
Trash Bit
Tag for 64(R2)
Tag for 16(R2)
Data
Trash
L4S
L3S
L2S
L1S
830
780
Ld R5, 64(R2)
820
Valid Bit
Trash Bit
R7S
R6S
Data
R5S
Trash
R4S
Trash
R3S
Data
R2S
Trash
R1S
R0S
Tags
Valid Bit
Trash Bit
Tag for 64(R2)
Tag for 16(R2)
Data
Trash
L4S
L3S
L2S
L1S
830

Figure 15

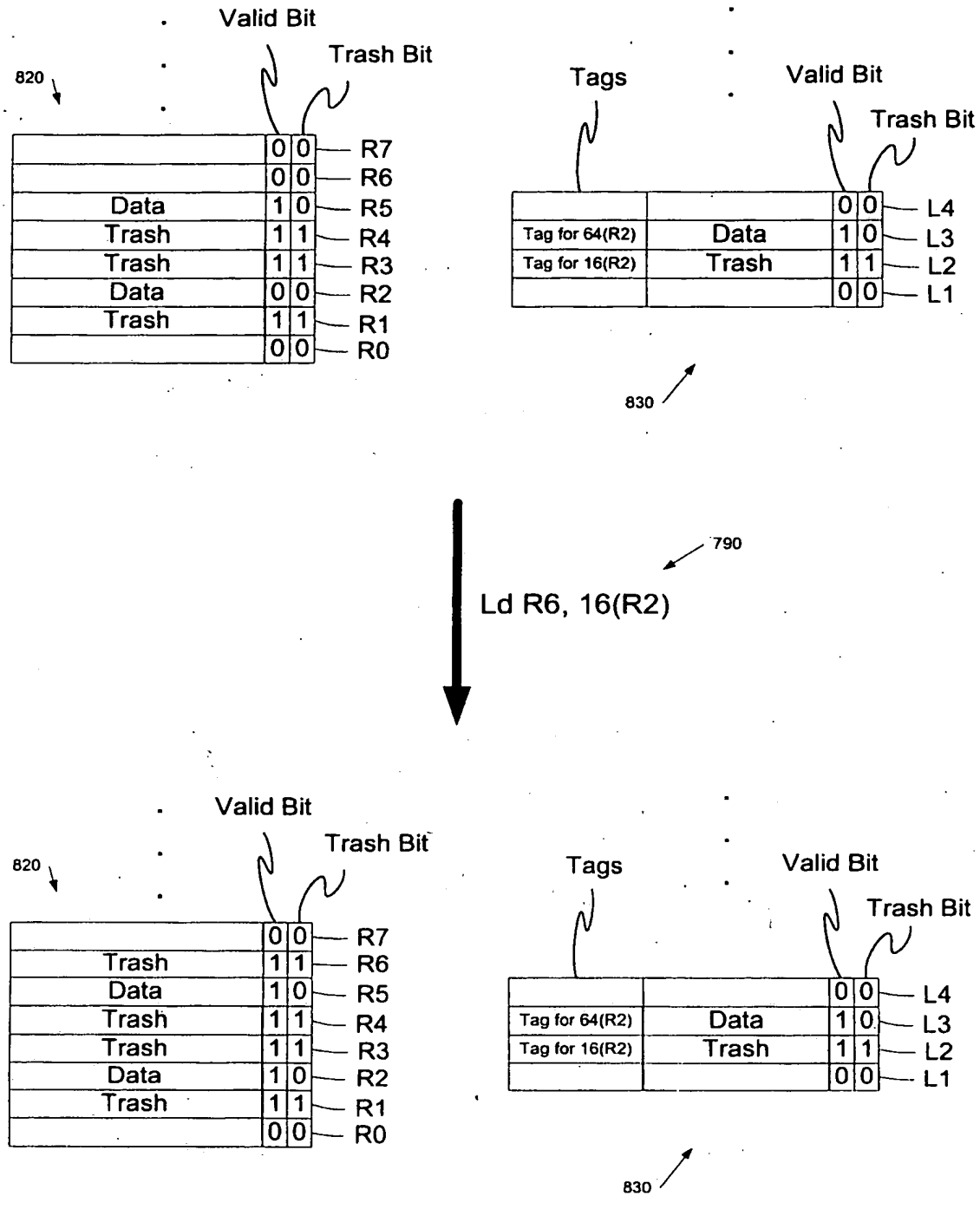
Valid Bit
Trash Bit
820
R7
R6
Data
R5
Trash
R4
Trash
R3
Data
R2
Trash
R1
R0
Tags
Valid Bit
Trash Bit
Tag for 64(R2)
Tag for 16(R2)
Data
Trash
L4
L3
L2
L1
830
790
Ld R6, 16(R2)
Valid Bit
Trash Bit
820
R7
Trash
R6
Data
R5
Trash
R4
Trash
R3
Data
R2
Trash
R1
R0
Tags
Valid Bit
Trash Bit
Tag for 64(R2)
Tag for 16(R2)
Data
Trash
L4
L3
L2
L1
830

Figure 16